

Emulators



6502 special

Simon Goodwin unearths seven more emulators for Commodore eight bit micros.



Your coverdisc contains *VICE*, *XC16* and *YAPE* sources and code, *Minus4* and *PfauZeh* archives, exclusive updates of *YAPE* and *XC16*, *cbmconvert*, classic Jeff Minter software, and other Commodore-related goodies.

Commodore entered 1985 with five micros, the portable *SZX-64*, (left) the seriously-styled *8296*, peak of the *Pets* (centre), the *Plus 4* (front), *C16* (top right) and classic *C64*, bottom right (see *LXF15*).

This month we have a tale of nice and nasty keyboards, custom chips, rude names in translation, and Commodore's determination to wring every drop out of the 6502. The C64, NES, Apple II and BBC Micro were famous users of the 6502 processor, as noted in past columns. Mos Technologies 6502s were at the core of Commodore's VIC-20, Plus 4 and C16, and the first one-box home computer, the Commodore PET, introduced in 1977.

Heavy PETting

Commodore rose to prominence in the cut-throat calculator market of the 1970s, and despite a solid metal case and futuristic styling their PET had calculator-like aspects, like a built-in screen and a flat keyboard on a rectangular rather than staggered grid. The former made it the first all-in one computer, with a far better display than the TV-based micro rivals, but the chicklet keyboard was a mistake that confounded typists. It was soon replaced with a typewriter one, but not before Sharp's MZ-80K copied it.

The PET displays a monochrome grid of 24 lines of 40 characters, preset from an eight by eight grid. Many special graphic characters were included in the ROM, and represent cursor-moves and similar display functions in listings. The inbuilt BASIC was a slightly-patched 8K interpreter, bought outright

from Microsoft before Balmer and Gates caught onto the idea of selling the same thing over and over. The built-in cassette drive loaded programs at 300 baud, but the 5.25" floppy disk drive, connected to the IEEE-488 parallel user port, was capable of shifting several kB per second, with luck and a following wind.

PET stood for **P**ersonal **E**lectronic **T**ransactor, upsetting Philips, who claimed it as a trademark, and Francophones were surprised to encounter a computer called a 'fart'. Original PETs came with 4K or 8K RAM, with spare sockets drilled out to discourage DIY upgrades. The 6502 processor was designed, like the PET itself, by former Motorola man Chuck Peddle, who peddled his chip factory to Commodore, giving them a head-start in custom chips which stood them in good stead for 15 years.

One emulator for Linux mimics many PET models, with variations accommodated by command line switches and menu options. *XPET* is part of the *VICE* package, and emulates the original PET 2001, and the later 3000, 4000 and 8000 models, with 80 column displays and up to 96K RAM.

Commodore's last 8-bit business computers were the 600 and 700 series, emulated by *CBM2*. These had larger, separate screens, 128K or 256K of bank-switched RAM, expandible to 1024K as their 2MHz 6509 processors combined an 8-bit data bus with 20-bit addressing. The unreleased 630 and 730, meant to have Z80 or 8088 co-processors, are not emulated.

VIC(s)tory

Commodore's VIC-20 was introduced in 1980, and sold millions worldwide over the following decade thanks to a good keyboard, passable sound and fast though chunky colour graphics. With just 23 lines of 22 characters on the TV screen, the small number of character cells in a complete display meant coders could obtain fast pixel-addressable colour graphics by redefining characters, when rival systems were stuck with preset fonts or mono bitmaps.

The name VIC-20 was not auspicious. The 20 refers vaguely to internal memory, but the majority of that is fixed ROM code,



derived from the earlier PET – which lacked specific support for the sound and graphics. The letters stand for **V**ideo **I**nterface **C**hip – the C64 equivalent is labelled VIC-2 – but sounds rude in countries which pronounce the letter V as an F, so the machine was known as a VC20 (Volks Computer) in continental Europe.

Only 3K of the 5K RAM is available for the user, but users soon added more – first 3K to fill a gap at address 1024 in the memory map, giving 8K vital for most commercial releases, then expansion in 8K steps up to address 32768 where the internal ROMs lurked. Further RAM or ROM cartridge expansion can stuff another 8K hole at around 40K.

Plus 4 and C16

The successful C64 reused the VIC-20 case with revamped silicon. We tested ten Linux-friendly emulations of that system a year ago (*LXF15*). Commodore followed that with low-cost variations, mass-produced in millions in the late eighties.

The Commodore 16, Plus 4 and related machines were easier to program but failed to repeat the success of the C64. By the time they came out their minimalist 6502 processors were no match for the newer 16-bit systems, though they attracted healthy software support, and now there's a choice of Linux emulators for both C16 and Plus 4.

The C16 was an economy version with 16K RAM, but graphics boosted from 16 to 121 colours. The TED chip that did this also lacked sprites. ROM cartridges and special versions of simpler C64 games were produced for the C16, and for a while it was well supported with budget cassettes, but the addition of extra ROM holding an editor/assembler and much-improved BASIC was not enough to bring it out from under the C64's shadow.

Transfers

Data resurrection

You can transfer programs via the real VIC's user port and a cable to your parallel port, although you need both machines in the same room to do this, and must type in a short program at each end to manage the transfer.

Another neat way to move programs is via the 1541 disk format, given some VIC software already on old CBM diskettes. The venerable C64 drive was developed for the VIC-20, and can be connected to Linux and accessed with PD handlers written for C64 emulators. *cbmconvert* converts programs in the raw '.prg' format to the '.p00' format XVIC needs, which has a 24 byte header:

```
cbmconvert -n -P vic/traxx.prg
```

This creates *trax.p00* in the current directory. You can then load this into XVIC by supplying the path name after the emulated drive number, e.g. **xvic -fs8 ~/cbm** and loading the file into BASIC:

```
LOAD "TRAXX",8,1 RUN
```

Alternatively you can assign a directory or disk image to a drive number (device 8 to 11) using the menus. As on the C64, which used the same serial peripheral interface and drives, other device numbers are used to communicate with printers, serial ports and other peripherals.

cbmconvert compiles readily on Linux if you unpack the archive and type **make unix**. *cbmconvert* can handle disk images and many other formats, but gets confused by upper-case file names and extensions in programs compiled on lesser systems, so you may need to rename input files.

Another attempt to cut costs brought forth the obscure C116, a re-packaged C16 with the typewriter keyboard supplanted by wobbly calculator-style keys. A VIC-10, based on the VIC20 hardware and PET 2001 keyboard, surfaced in Japan.

Plus 4

Plus 4s have 64K memory like C64s, and the extra ROM of the C16, all packaged in a slimline black box. The name comes not from Edwardian trousers but the four productivity programs (rather feeble 'business' software) in an extra 32K of bank-switched ROM, first known as Three Plus One.

XPET and XCBM2

XPET emulates several models of the all-in-one PET, the *VICE* emulator options let you switch through the early years of PET development, with support for the buggy 'old ROM' and tape files as well as disk images.

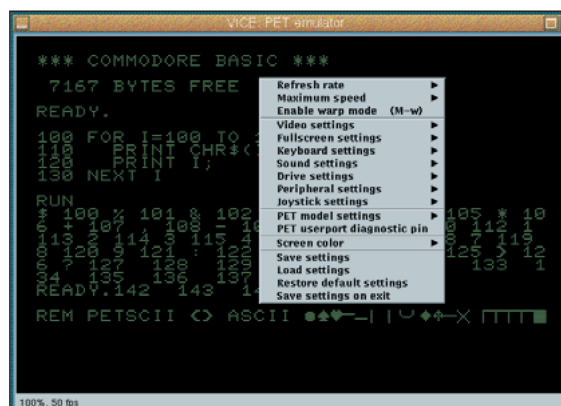
The CBM2 computer was also known as the C610. Again *VICE* emulates several models from the range, though not the later multi-processor versions. These are authentic emulators, but the firmware is basic in every sense, and mono character graphics limit applications. Unless you've used the originals, *XPET* and *XCBM2* are likely to leave you cold.

PfauZeh

VICE competes for the VIC-20 emulation crown with *PfauZeh*, which puns on German pronunciations of the letters V and C. *PfauZeh* stalled in 1998 at version 0.8 on Linux, though work on the Win32 version continues. It is supplied as over a megabyte of compiled binary, with ROMs, HTML files and an example game. *PfauZeh* ran on my Debian 2.2 installation once I reset the video depth to eight bit to provide access to palette hardware, and switched my usual *Blackbox* window manager for the minimalist *TWM*, freeing up unallocated colour for the emulator.

PfauZeh initially presents a titchy X window, though two pixels are allocated to every one on a VIC line to keep the aspect ratio about right. The **-hires** command line option gives a larger view, plotting four pixels in X and two in Y for each original one.

Screen updates default to 30Hz – one for every two fields



XPET, showing a *VICE* menu and PETSCII characters.

drawn by the original hardware – but you can crank this up to 60Hz for the default NTSC video mode, emulating the original 6560 VIC chip, and from 25 to 50 Hz for the 6561 PAL version. Three menu options select the video format, frame rate from 1:1 to one in six, and nine or 28 lines of overscan at the top of the screen. Sliders pick the original 5K RAM, or 3K, 8K, 16K or 24K expansion, with a toggle for extra RAM at address \$A000.

A delay knob slows emulation by wasting from zero to 500 cycles every time an instruction is fetched, but *PfauZeh* is hardly speedy. A 10,000 iteration FOR loop runs across two BASIC lines in about 16 seconds on my K6 with 60Hz display, and in 12 seconds at 30Hertz, about as slow as the original. ➤➤

On the web

This series has reported on more than 100 emulators already, and the results now nestle at <http://simon.mooli.org.uk/LXF>.

So far this site only contains early parts of the series, but it will eventually host the lot, plus material that didn't make it into the magazine.

Emulators

« A file menu will load or save BASIC and plug in or remove a ROM cartridge, with reset implied by removal. Another menu option saves display, CPU and memory configuration settings. The **-joy** emulation maps a Linux joystick to mimic a VIC one. The alternative keyboard joystick emulation uses the right **Ctrl** and cursor keys, or numeric pad arrows with **NumLock**, for directional moves. Two analogue paddles are emulated with a mix of cursor and numeric pad keys. The mouse can also simulate a lightpen, for those few VIC programs that expect one. The mouse pointer shows where the pen is pointing and any mouse button emulates the one on the pen.

PfauZeh emulation seems precise, with undocumented 6502 opcodes, interrupts and instruction timings correctly modelled, plus both VIA interface chips and all VIC graphics modes, including the 16 colours, screen position and raster beam registers. Video output is line based, rather than synchronised with whole fields or individual pixels, so it copes with programs that reprogram the VIC in horizontal flyback but not those few that rely on precise timings to tweak it during a line.

The HTML documentation is a good read, neatly but rather garishly written mainly in red or yellow on black. It contains no links and only one graphic, showing the original VIC keys and their PC equivalents. The author lists many programs known to be compatible, with advice on memory size, PAL and NTSC dependencies, and **SYS** commands to get programs running. **SYS** invokes a block of 6502 code on many Commodore machines by diverting the program counter to a given address.



Commodore's PET 2001 was the first all-in one personal computer, but the fiddly key grid was bad news for touch-typists.



***PfauZeh's Thriller* demo is mostly written in BASIC.**

Compatibility

If your files are intact *PfauZeh* should run almost anything you throw at it. Only two programs fail outright, though half a dozen others show minor timing or graphical glitches and one of the ones that doesn't work is OK in its NTSC incarnation and only bombs on PAL. Sound supports the eighties standard of three tones and one noise channel, converted to eight bit mono at 44.1 kHz, though I found this noticeably jerky on the games I tried, whatever the update options. You can run *PfauZeh* without sound.

Keyboard input is also erratic even on a 500 MHz K6, with

momentary hiccups in the display update as every few characters are entered into BASIC. The **F11** key or a menu option reset the VIC, **Esc** quits and **F2** and **F4** are shortcuts to BASIC **LOAD** and **SAVE** requesters, which use *Motif* file selection. I was not able to get save to work, despite checking file permissions.

The tab key stands in for Commodore's **RUN/STOP** button, and left **Alt** for the special **CBM** key. **Insert** generates a £ or Lire sign. Cursor movement is authentically but irritatingly achieved using **PageDown** and **PageUp** to move down and right respectively, with **Shift** to reverse the direction. **F12** acts like the VIC's **RESTORE** key and the letters, numbers, space bar, **Enter** and **Home** keys work as expected. Odd numbered function keys from **F1** to **F7** mimic the four original VIC function keys, with even numbered keys assigned to adjust the emulators frame rate or memory size.

Jerky sound and keyboard input spoil what is otherwise a comprehensive emulator, especially in its graphics. More refined host device access could make it attractive, and while Arne Bockholdt is no longer involved with Linux he is willing to help others who'd like to update it.

XVIC

XVIC is part of the *VICE* package, like *XPET* and *X64*. Like those it shows the frame rate and percentage of original CPU time in the bottom window border. It defaults to 50Hz and authentic 100% CPU speed but can manage over 400% and 200 frames per second if you take off the brakes, and ten times the original CPU speed at about half the authentic frame rate if you slew the options that way. On the same 8-bit X11 window with two for one pixel mapping and the *XVIC* default 50Hz display update it was as fast as *PfauZeh* with its half-NTSC 30Hz video, without the sound and keyboard hiccups, and four to eight times faster than the real thing.

XVIC benefits from *VICE* speed-ups like Warp mode, video caching and configurable sound sample rate. X display depths may range from 1 to 32 bits per pixel, with four drives emulated by **LOAD** and **SAVE** commands from BASIC, rather than menus that bypass the VIC operating system.

Speed limits are by percentage of full speed, and may far exceed 100 on a modern Linux box. Double size, double scan and full screen modes are controlled from the command line or two full sets of menus accessed by left and right mouse buttons.

Four RS232 devices with individual baud rate settings, normally connected to `/dev/ttyS0` and `/dev/ttyS1`, the Linux serial ports, the file `'rs232.dump'`, and a pipe to `'lpr'` for printing. Access to the VIC user port can also be diverted to a *VICE* RS-232 emulation, and four printer devices are similarly linked to files, Unix devices, or *VICE* IEC port emulation.

1541, 1571, 1581, 2031 or SpeedDOS disk drives can be emulated down to the level of the drive firmware, which is very accurate but costs a lot of time, or implemented by ROM patches that fool most software at lower overhead, and mapped to host directories either way.

Two joysticks and the 1351 proportional mouse can also be emulated by Linux peripherals. *XVIC* has impressive sound, well used in the Jeff Minter games on our cover disc, but struggles with some of the Llama-man's graphical jiggery-pokery, especially on menu screens where you may just get erratic flashing rather than Jeff's trademark beam synchronised psychedelia.

C16 and Plus 4

The only C16 emulator I found specifically for Linux was *xc16emu* version 0.5.4, written in 1998 for *Qt 1.3* or *1.4*, and I had to comment out some cosmetics to fit the C++ to the *Qt 2.3.0* on my Debian box. I replaced the caption in line 71 of *xc16emu.cc* with a literal string, removing lines 53 and 54 in *mainwindow.cc* and the body of the **about** and **info** methods there. The result is on your coverdisc.

Video and CPU ran at about double speed on my 500MHz K6 and Matrox G400, using the default 1:1 pixel window. Scaling the screen up to Zoom 3 or Zoom 4 brought authentic speed and full-screen. *xc16emu* has been tested on MIPS and Alpha.

Load the four simple demos in the BASIC subdirectory with **DLOAD "BASIC/file.BAS"**, press TAB to stop them. Once tweaked, with ROM images from funet, *xc16emu* offers a pretty GUI, let down by imprecise CPU timings and lack of sound.

Plus 4 emulators

The *Minus4Java* applet emulates the Commodore Plus 4 and C16 slowly but well enough to run many games, redirecting cursor key-presses to the emulated joystick. The implementation as an Applet means that it cannot save any results. A 500MHz Celeron, Netscape *JRE1.2*, RH 7.0 ran at 50Hz window updates.

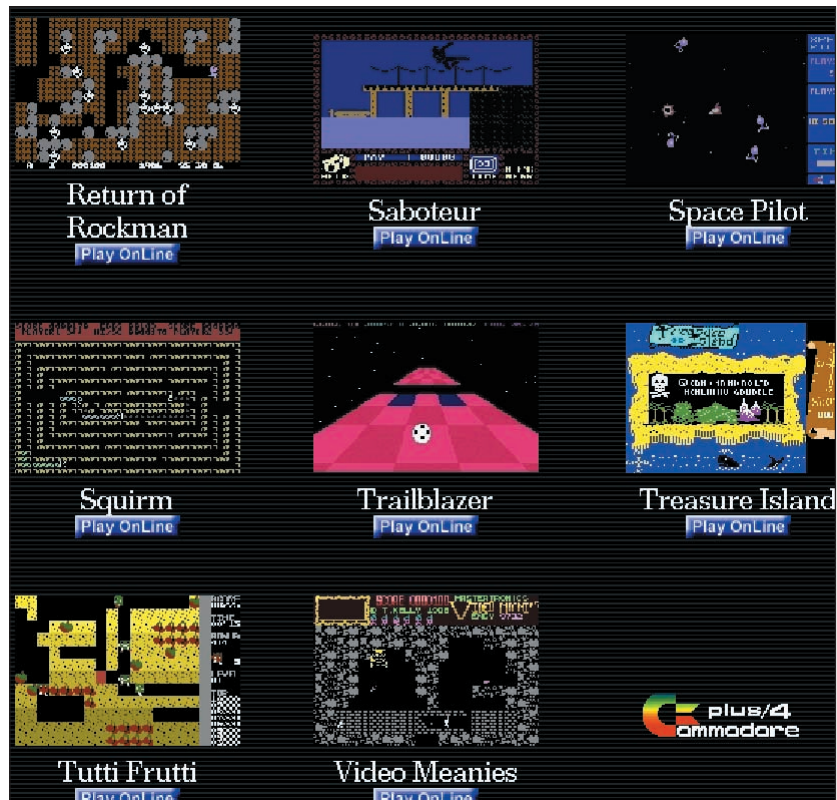
YAPE

Yet Another Plus/4 Emulator, written by LXF reader Attila Grosz. Attila lost his Mandrake set-up recently and has been awaiting an upgrade before re-starting work. The GPL'd *YAPE* is written in C++, and requires *LibSDL 1.2.1* or later.

YAPE compiled without demur on my Debian 2.2 test system, with *SDL 1.2.3* installed. Just type **make** then **./yape** to get a familiar Commodore BASIC screen in an X11 window. *SDL* console messages warned it was unable to find the desired 8-bit unsigned mono audio output at 44,100Hz, but settled for a close equivalent mode supported by my SoundBlaster Live. After the command you can specify the name of a TAP tape image or a PRG file holding program bytes ready to be loaded directly to memory. Files load from the '~/.yape' subdirectory.

Once *YAPE* is running, the first few function keys give quick access to BASIC commands like **GRAPHICS**, **DLOAD**, **DIRECTORY** and **HELP**, as on a real Plus 4. **F5** and **F6** simulate cassette drive **PLAY** and **STOP** keys. **F7** saves screenshots in the bloated BMP format – *SDL* makes some things too easy! The start screen comes out as a 143,350 byte BMP, shrinking to 1819 bytes once *GIMP* converts it to PNG. **F10** saves current settings to your home directory. **F11** with optional **Ctrl** or **Shift** triggers three types of +4 reset, and **F12** and **Esc** quit.

Left **Alt** with **G**, **J**, **I** and **S** toggle a display of the frame rate above the main window, joystick emulation with Linux cursor keys,



the emulated joystick port, and a frame rate limiter, clamped to 50Hz rather than 50MHz as mistakenly documented!

Sample Plus 4 games online in *Minus4* at plus4.emucamp.com

Key points

YAPE keyboard handling is conceded to be a bit shaky, and depends on a table mapping *SDL* codes to Plus 4 matrix positions. **End** sends the £ or Lire key code, but I found the zero keys did not work at all with my US PC keyboard. After some fiddling with the source and tentative key-pressing I found out that **0** is *SDL* key **\$30** and that the key to the left of **1**, with a back tick and tilde on it, generated the zero code, which is **\$60** to *SDL* (and ASCII) and 28 for the +4. That key is not part of the 64 key Plus 4 matrix, so the emulator is usable once you know to press that when you want a zero. Arguably this is more logical than the normal arrangement of having 0 after 9, but like most people I've got used to the latter scheme, so I changed the first entry on line 29 of keyboard.cpp from 38 to 28 and got the normal zero key working, as well as the one before **1**. Even with a weird keyboard you should be able to remap the emulated keys this way. Tweaked source and a compiled version of the emulator is on your coverdisc with the author's original archive. [LXF](#)

Links

A web of Commodore sites:

C16 Deutschland: www.c16.de

C for CBMs: www.acc.umu.se/~arvid/cc65_mirror

Link cable: www.lb.shuttle.de/puffin/cbm4linux

LXF emus online: <http://simon.mooli.org.uk/LXF>

Minus4: www.javalemmings.com/minus4/files

PfauZeh home: www.classicgaming.com/pfauzeh

Plus 4 online: www.plus4.org

VIC-20 ammunition: <ftp://ftp.funet.fi/pub/cbm/vic20>

VICE home: <http://viceteam.bei.t-online.de>

XC16: <http://ibiblio.org/pub/Linux/system/emulators/>

YAPE: www.keepitretro.com/plus4